41

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/892,329	06/27/2001	Marcus Peinado	MSFT-164268.1	1912

41505 7590 07/07/2006

WOODCOCK WASHBURN LLP (MICROSOFT CORPORATION)
ONE LIBERTY PLACE - 46TH FLOOR
PHILADELPHIA, PA 19103

EXAMINER	
SHIFERAW, ELENI A	
ART UNIT	PAPER NUMBER
2136	

DATE MAILED: 07/07/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/892,329	**Applicant(s)** PEINADO ET AL.	
	Examiner Eleni A. Shiferaw	**Art Unit** 2136	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *17 April 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *15,16,20-27,31,32 and 36-43* is/are pending in the application.

4a) Of the above claim(s) *1-14,17-19,28-30 and 33-35* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *15,16,20-27,31,32 and 36-43* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

Response to Arguments

1. Applicant's arguments and amendments with respect to claim amended all independent claims to include "instantiating the application only after the security kernel has authenticated such application" and dependent claims have been considered but are moot in view of the new ground(s) of rejection.

2. Applicant's argument regarding neither the Vu reference nor the Ginter reference teaches or even suggests transitioning between modes by way of a CPU reset as claimed in dependent claims 16 and 31 are not persuasive. Because Vu discloses the transitioning between modes during a CPU power-on (see, Vu fig. 1, col. 4 lines 12-39 and col. 5 lines 18-20).

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 15-16, 20-27, 31-32, and 36-43 are rejected under 35 U.S.C. 103(a) as being unpatentable over Vu et al. (Vu, Patent No.: US 6,557,104 B2) in view of Ginter et al. (US 5,892,900) and Mirov et al. (Mirov, USPN 6,138,236).

As per claim 15 and 31, Vu teaches a method/medium for a secure processor to instantiate and authenticate a secure application thereon by way of a security kernel, the method comprising:

entering a preferred mode where a security key of the processor is accessible (Vu col. 5 lines 25-35; *enters a secured mode to access security key/crypto key*);

instantiating and running a security kernel, the security kernel:

accessing the security key (Vu col. 5 lines 35-36);

applying the accessed security key to decrypt at least one encrypted key for the application (Vu col. 5 lines 35-40);

storing the decrypted key(s) in a location where the application will expect the key(s) to be found (Vu col. 6 lines 65-col. 7 lines 11); and

authenticating the application on the processor (Vu col. 5 lines 36-40, and col. 7 lines 7-11); and

entering a normal mode from the preferred mode after the security kernel authenticates the application (Vu Fig. 2 No. 25 and col. 5 lines 42-47),

where the security key is not accessible; wherein the security kernel allows the processor to be trusted to keep hidden the key(s) of the application (Vu col. 4 lines 63-col. 5 lines 9); and

wherein the security kernel employs the accessed security key during the preferred mode to authenticate/verify the application prior to instantiating thereof (Vu col. 5 lines 35-40).

Vu teaches all the subject matter as cited above. Vu fails to explicitly teach:

erasing data in the cache of the processor when entering preferred mode such that any data previously stored in the cache is not available to interfere with preferred mode operations; and erasing data in the cache of the processor when entering normal mode such that any sensitive data in the cache from preferred mode operations is not available during normal mode from such cache as argued.

However Ginter et al. discloses the argued subject matter as well-known as follows:

wherein the processor has a cache, the method further comprising:

erasing data in the cache of the processor when entering preferred mode such that any data previously stored in the cache is not available to interfere with preferred mode operations (col. 75 lines 18-29; *at the end of secure processing/when exiting the secure mode/disabling SPU mode, contents of all registers and other temporary storage/cache locations used within secure memory are destroyed*); and

erasing data in the cache of the processor when entering normal mode such that any sensitive data in the cache from preferred mode operations is not available during normal mode from such cache as argued (claim 53; *mode switch circuitry of secure processor that switches normal mode to secure mode and/or secure mode to normal mode, deletes information stored in temporary storage/cache locations that are outside the secure memory upon detection of the secure processing unit is about to transition into secure mode*).

Therefore it would have been obvious to one having ordinary skill in the art at the time of the invention was made to combine the teachings of Ginter et al. with in the system of Vu because they are analogous in secure method of distribution of content/application to end users (col. 315 lines 43-col. 326 lines 48, and col. 73 lines 56-col. 76 lines 36). One would have been motivated to incorporate the teachings of erasing the cache data when switching modes within the system of Vu to prevent execution based on "mixed" secure and non-secure instructions and to prevent access carried over from outside mode, cache coherency control access, to insure that the microprocessor is controlled entirely by instructions carried within or derived from the secure memory, and/or to prevent other CPU operations/instructions from exposing the contents of secure memory (col. 74 lines 29-48).

Vu and Ginter et al. do not explicitly disclose instantiating the application only after the security kernel has authenticated such application.

However Mirov discloses the well-known application initialization after authenticating such application (see col. 3 lines 56-col. 4 lines 55 and claim 9).

Therefore it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the teachings of Mirov within the combination system of Vu and Ginter because it would authenticate the integrity of software before executing the boot up sequence for a computer system (col. 2 lines 7-64). One would have been motivated to do so because it would authenticate an application to insure that it is in fact the original application and/or not altered.

As per claim 25 and 41, Vu teaches a method/medium for a secure processor to instantiate one of a plurality of available secure applications thereon by way of a security kernel, the method comprising:

setting a chooser value to a value corresponding to a chooser application upon power-up (Vu col. 4 lines 12-39, col. 5 lines 1-4 and18-20, col. 1 lines 11-col. 2 lines 49 and abstract; ***user providing PIN/confidential information and... application to gain access to... online electronic commerce...***);

entering a preferred mode upon a power-up CPU reset and instantiating the security kernel, the security kernel determining that the chooser value corresponds to the chooser application and therefore authenticating same, the chooser application being instantiated (Vu col. 4 lines 52-col. 5 lines 8, and col. 5 lines 36-40);

entering a normal mode after the chooser application is instantiated and leaving same to run, the chooser application presenting the plurality of available applications for selection by a user (Vu col. 5 lines 40-44 and fig. 2 No. 25);

receiving a selection of one of the presented applications to be instantiated (Vu col. 5 lines 32-40);

setting the chooser value to a value corresponding to the selected application (Vu col. 4 lines 12-39 and page 5 lines 18-20);

re-entering the preferred mode upon an executed CPU reset and instantiating the security kernel, the security kernel determining that the chooser value corresponds to the selected application and therefore authenticating same, the selected application being instantiated (Vu col. 4 lines 52-col. 5 lines 47, col. 5 lines 36-40 and ***col. 3 lines 14-18; each time and/or when ever requested the processor enters the secure mode and entering to secure mode and to normal mode and vise versa is clearly taught by Vu***);

entering a normal mode after the selected application is instantiated and leaving same to run (Vu col. 5 lines 40-44 and fig. 2 No. 25);

wherein the security kernel allows the processor to be trusted to keep hidden a secret of the chooser application and a secret of the selected application (Vu col. 4 lines 63-col. 5 lines 29).

Vu teaches all the subject matter as cited above. Vu fails to explicitly teach:

erasing data in the cache of the processor when entering preferred mode such that any data previously stored in the cache is not available to interfere with preferred mode operations; and

erasing data in the cache of the processor when entering normal mode such that any sensitive data in the cache from preferred mode operations is not available during normal mode from such cache as argued.

However Ginter et al. discloses the argued subject matter as well-known as follows: wherein the processor has a cache, the method further comprising:

erasing data in the cache of the processor when entering preferred mode such that any data previously stored in the cache is not available to interfere with preferred mode operations (col. 75 lines 18-29; *at the end of secure processing/when exiting the secure mode/disabling SPU mode, contents of all registers and other temporary storage/cache locations used within secure memory are destroyed*);

erasing data in the cache of the processor when entering normal mode such that any sensitive data in the cache from preferred mode operations is not available during normal mode from such cache as argued (claim 53; *mode switch circuitry of secure processor that switches normal mode to secure mode and/or secure mode to normal mode, deletes information stored in temporary storage/cache locations that are outside the secure memory upon detection of the secure processing unit is about to transition into secure mode*); and

Ginter also teaches a chooser value and plurality of chooser applications wherein an end-user selecting a content/application, performing an authentication, analyzing usage and rights and providing a content/application to the user (col. 315 lines 43-col. 317 lines 42).

Therefore it would have been obvious to one having ordinary skill in the art at the time of the invention was made to combine the teachings of Ginter et al. with in the system of Vu because they are analogous in secure method of distribution of content/application to end users (col. 315 lines 43-col. 326 lines 48, and col. 73 lines 56-col. 76 lines 36). One would have been motivated to incorporate the teachings of erasing the cache data when switching modes within the system of Vu to prevent execution based on "mixed" secure and non-secure instructions and to prevent access carried over from outside mode, cache coherency control access, to insure that the microprocessor is controlled entirely by instructions carried within or derived from the secure memory, and/or to prevent other CPU operations/instructions from exposing the contents of secure memory (col. 74 lines 29-48).

Vu and Ginter et al. do not explicitly disclose instantiating the application only after the security kernel has authenticated such application.

However Mirov discloses the well-known application initialization after authenticating such application (see col. 3 lines 56-col. 4 lines 55 and claim 9).

Therefore it would have been obvious to one having ordinary skill in the art at the time of the invention was made to modify the teachings of Mirov within the combination system of Vu and Ginter because it would authenticate the integrity of software before executing the boot up sequence for a computer system (col. 2 lines 7-64). One would have been motivated to do so because it would authenticate an application to insure that it is in fact the original application and/or not altered.

As per claims 16 and 32, Vu teaches the method/medium wherein entering the preferred mode comprises entering the preferred mode upon a CPU reset (Vu col. 4 lines 12-39 and col. 5 lines 18-20).

As per claims 20 and 36, Vu teaches the method/medium wherein the security kernel performs a hash/MAC (message authentication code) over at least a portion of the application and then compares the hash/MAC to a hash/MAC corresponding to the application (Vu col. 7 lines 1-11).

As per claims 21-22, and 37-38, Vu teaches the method/medium wherein the security key of the processor is a symmetric key and the application is instantiated from a code image including a main body and a header including:

KCPU (KMAN)	KMAN encrypted according to KCPU
MAC (main body, KMAN)	message authentication code of the main body under KMAN
KMAN (KCODE)	KCODE encrypted according to KMAN

where KCPU is the security key, KMAN is a device key of the portable device independent of the security key, and KCODE is the secret of the application, and wherein the security kernel applying the accessed security key to decrypt at least one encrypted key for the application comprises:

applying KCPU to KCPU (KMAN) to produce KMAN (Vu col. 6 lines 64-65; encrypted encryption key);

computing MAC (main body, KMAN) (Vu col. 5 lines 32-40 and col. 7 lines 1-11);

comparing the computed MAC to MAC (main body, KMAN) from the header to determine if the code image has been changed (Vu col. 5 lines 32-40 and col. 7 lines 1-11); and

if the MACs match, applying KMAN to KMAN (KCODE) to produce KCODE (Vu col. 5 lines 32-40 and col. 7 lines 1-11).

As per claim 23, and 39, Vu teaches the method/medium wherein the security key of the processor is a private key of a public key--private key pair and the application is instantiated from a code image including a main body and a header including:

public key (KCODE)	KCODE encrypted according to the public key

where KCODE is the secret of the application, and

wherein the security kernel applying the accessed security key to decrypt at least one encrypted key for the application comprises applying the security key as the private key to public key (KCODE) to produce KCODE (Vu col. 7 lines 31-35).

As per claims 24 and 40, Vu teaches the method/medium wherein the security key of the processor is a private key of a public key-private key pair and the application is instantiated from a code image including a main body and a header including:

public key (HASH (main body), KCODE)	Hash of the main body and KCODE, both encrypted according to the public key

where KCODE is the secret of the application, and

wherein the security kernel applying the accessed security key to decrypt at least one encrypted key for the application comprises:

computing HASH (main body) (Vu col. 5 lines 32-40, col. 7 lines 1-11 and lines 31-35);

applying the private key to public key (HASH (main body), KCODE) to produce HASH (main body) and KCODE (Vu col. 5 lines 32-40, col. 7 lines 1-11 and lines 31-35);

comparing the computed HASH to the produced HASH to determine if the code image has been changed (Vu col. 5 lines 32-40, col. 7 lines 1-11 and lines 31-35); and

if the HASHs match, employing the produced KCODE as appropriate (Vu col. 5 lines 32-40, col. 7 lines 1-11 and lines 31-35).

As per claims 26 and 42, Vu teaches the method/medium further comprising setting the chooser value to the value corresponding to the chooser application upon the selected application being authenticated by the security kernel, wherein upon execution of a CPU reset, the security kernel determines that the chooser value corresponds to the chooser application 72c and therefore authenticates same (Vu col. 4 lines 12-39 and col. 5 lines 18-20).

As per claims 27 and 43, Vu teaches the method/medium further comprising storing the chooser value in a memory location not affected by a CPU reset so that the stored chooser value is available after same (Vu col. 5 lines 11-23).

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Angelo et al. 6,581,162 B1 discloses user request for secure communication and user entering a key/PIN and authenticating entered data to boot the user operating system upon power-up cycle.

5. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eleni A. Shiferaw whose telephone number is 571-272-3867. The examiner can normally be reached on Mon-Fri 8:00am-5:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ayaz R. Sheikh can be reached on 571-272-3795. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

E.S.
6/26/06

AYAZ SHEIKH
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100